DOCUMENT RESUME

ED 280 282 FL 016 543

AUTHOR Eastment, David
TITLE Teacher Resistance to CALL.
PUB DATE Apr 86
NOTE 9p.; Based on a paper presented at a conference of the International Association of Teachers of English as a Foreign Language (Brighton, England, April 1-4, 1986).
PUB TYPE Viewpoints (120) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS College Students; *Computer Assisted Instruction; *Computer Software; Courseware; Databases; Educational Improvement; Educational Needs; *English (Second Language); Higher Education; *Language Teachers; *Negative Attitudes; Second Language Instruction; *Teacher Attitudes; Teacher Role; Word Processing

ABSTRACT

Despite the evolution of software for computer-assisted language learning (CALL), teacher resistance remains high. Early software for language instruction was almost exclusively designed for drill and practice. That approach was later replaced by a model in which the computer provided a stimulus for students, most often as a partner in games. However, these games are essentially limited exercises and consequently, student recall of the content is limited. They are also difficult for most teachers to integrate into the curriculum, and they actively erode the teacher's role by challenging the student to find a single correct answer. Other, more general factors inhibit teachers' use of existing software, including the time needed to become familiar with it, lack of standardization, and students' pre-existing familiarity with the programs. Rather than producing more of the same kinds of materials, the language profession should turn its attention to applications software such as word processors, spreadsheets, database communications programs, graphics packages, and desk-top publishing. They are distinguished by the need to learn only one program, the fact that they are real, not only realistic, their portability, and the genuine role given the teacher. The programs are tools, not teachers, and are gaining substantially in interest in education. (MSE)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED280247

TEACHER RESISTANCE TO CALL

David Eastment

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

✓ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

D. Eastment

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Paper presented at a conference of the International Association of Teachers of English as a Foreign Language (Brighton, England, April 1-4, 1986)

FL 016543

BEST COPY AVAILABLE

TEACHER RESISTANCE TO CALL

This article is based on a paper given at IATEFL, Brighton in April 1986.

We started to introduce Computer Assisted Language Learning in my College some four years ago. There was a great deal of enthusiasm; but this rapidly waned as teachers realized that we only had four programs which would run on our machines! The answer, we thought, was to build up our software base as quickly as possibly; and this is what we set about doing. Today, we have well over a hundred programs (1), and hundreds of datafiles available at every level for text and vocabulary activities. But although every teacher is given some initial training in CALL, there is still a clear division between those who use the computers regularly and those - probably around 50% - who prefer not to use them at all. In this article, I would like to examine some of the reasons for this teacher resistance (incidentally, it is perhaps significant that we have scarcely ever had the slightest student resistance!), and suggest some ways in which it can be overcome.

In the sixties and seventies, CALL programs were overtly constructed to present a single item of language to the student, who would typically be seated alone at a terminal to a large and expensive mainframe. The program would drill the item again and again until no more mistakes were made, and then move on to the next. Even in the eighties, this type of CALL persists most notably in the United States (2). Its popularity is scarcely surprising, since it reflects the way that computer programmers are taught to write code: break down each task into distinct sub-tasks, and execute them one by one. But the "drill-and-practice" approach is scarcely consonant with modern theories of language learning, and has been replaced, thanks in part to the work of Higgins and Johns (3), by a model in which the computer provides the stimulus for a group of students clustered around the screen. Rather than being a "magister", the computer is cast in the role of games player, or partner, or opponent.

This "ludendic" model has been greatly in vogue over the last few years, since it re-evaluates, even reverses the role of the computer in the classroom. The software is typically task-orientated, and challenges the students to find anagrams, complete crossword puzzles, unjumble a text, fill in the blanks, guess the title of a partly-concealed passage, etc. It is popular with publishers since it is relatively easy to produce software of this sort; and it is popular with students, since at their best, these programs can be motivating and enormous fun. But they carry with them every real disadvantages.

They are essentially limited exercises. There is a finite number of manipulative changes you can ring on a text; and an exercise that is fun the first few times can rapidly become merely tiresome. Often, too, their "challenge" element is misplaced. Students can easily get absorbed in the immediate challenge of finding the next missing word, for example, at the expense of any global comprehension of the text. I have conducted a number of informal surveys on students coming out of our computer room and been astonished by the very limited recall they showed of the text they had only minutes before been working with.

Software of this type tends, too, to be remarkably difficult for most teachers to integrate into the rest of their teaching schedule. The computers are used either as a fringe "Friday afternoon" activity or, at the other extreme, as a one-off exercise in which the software is certainly the primary focus of attention but seems to be used for no other reason than because it is there! Now I do accept that this is to a certain extent a methodological problem. It is a valid criticism of many institutions that they have devoted enormous attention to hard-ware configurations and providing large software libraries, and very little attention to how the programs should actually be used. It is true that if enough attention is paid to methodology certain programs can be slotted in effectively to the rest of the teaching programme (4). What I am arguing is that this is done in spite of the software, rather than because of it.

./

What is perhaps even more serious is that a great deal of this software can actively erode the teacher's role. In part, this is due to the beguiling nature of the VDU - whatever is happening on-screen is almost certainly more interesting than what the teacher has to say! But more importantly, it reflects the fact that however much we may have redefined the computer's role, most programs only know one "right answer", and the challenge to students consists exclusively in them finding it. Generally, the computer program will "help", if required, by providing the first letter of a word, or giving the number of letters in it. As this process goes on, there is very little the teacher can do, beyond wandering around the room monitoring what is going on at each machine. And it is not a role that all teachers are comfortable in!

Again, there are methodological ways around this: getting students to author texts themselves, for example, is a common one. But the fact is that most CALL software doesn't actually require a teacher to be present to work effectively.

All the foregoing is relevant to ELT software, but there are other, more general factors which may account for many teachers' unwillingness to use micros with their classes.

One of these is time. Getting familiar with software is immensely time-consuming. A learning program cannot be skimmed through like a book or fast-forwarded like a video (although there is no reason why publishers should not provide a "skim-facility" of selected frames, to give the user at least a flavour of the contents of the package). It needs to be gone through in real-time, and each of the various branches explored. In the case of a complex simulation, this may take several hours.

The problem is further compounded by the apparent inability of software houses to agree on any sort of standardization. Thus, while in one program you may continue by pressing 'C', in another it might be enough to press any key; in a third, only the Space Bar will do; in a fourth, it just has to be the Return Key.

./

A final consideration that can be off-putting for many teachers is the fear that any programs they use will already be familiar to students. This is especially a problem for networked systems, which tend to be organized to provide as open an access as possible. But, as with other material, teachers tend to feel more confident if they can be sure that their class will not have used it before.

I do not want to appear too extreme in my criticism of existing EFL software. Much of it can be useful and rewarding, particularly on short courses, or when the teacher is prepared to spend a great deal of time tackling methodological problems. A great deal is very valuable in self-access, accommodating as it often does a wide range of learning styles and allowing the student to work at his own pace and as often as he wishes. And a substantial number fall outside the ambit of my criticism since they do not "challenge" the user to provide a "right" input. Nevertheless, I do still feel that we have too much software in ELT, and that a great deal of it is trivial and unrewarding (at least for the teacher) and does not repay the investment in time required to familiarize oneself with it.

So what is the way ahead? It seems to me that we should not reject any of the manipulative programs out of hand, any more we should entirely neglect the drill-and-practice software. While they may not be everyone's taste, they still have something to offer for certain students and in certain situations. But rather than devoting our energies to producing more of the same, perhaps we would be better advised to turn our attention to an entirely different area: applications software.

By "Applications software" I mean such things as word processors, spreadsheets, databases communications programs, graphic packages, desk-top publishing etc. What distinguishes them, as far as the teacher is concerned is:

1. The time factor. Only one program needs to be learnt, rather than having to come to grips with the heterogeneous protocols of several dozen. Having familiarized himself with say, how the spreadsheet operates, the teacher has more time to devote to pedagogy (and the learners!)

2. Using applications software is a real activity, rather than a realistic one. Even the most sophisticated simulations can only approximate to the reality whereas databases etc are the very stuff of the world outside. The student is motivated by the activity itself, and not just by the novelty of the activity.

3. Applications software is portable. Unlike a cloze-exercise or speed-reading package which will run on an Apricot but not on an Apple, on a BBC Micro but not on an IBM PC, virtually all micros have applications software which may differ in name, power and sophistication but essentially does the same thing. The lessons learnt through using one package in a certain way can thus be transferred to any similar package.

4. The teacher is provided with a genuine role. These programs are tools, not teachers (or games players) and make no attempt to evaluate user input. There is no right answer, and the teacher is therefore able to take on the role of "knower" or "source".

I would like to be able to say that the increasing use of 'real-world' software has completely overcome teacher resistance to CALL; but it is still much too early to draw such a conclusion. Some uses are developing rapidly: word-processing is the clearest example, with communications packages such as teletext emulators or the ability to access parts of PRESTEL and CEEFAX from all stations a close second. There

./

is a substantial interest in database work, too, although this is hampered at the moment by the relatively primitive machines we have available. Spreadsheets have thus far proved more intractable: might this be because most EFL teachers are innumerate?

Writing in 1984, John Self observed: "..The widespread belief that we already have good educational software or, if not, that it is just round the corner, seems to me entirely unfounded (.....). The case for computer-assisted learning using present software seems very weak" 5.

In 1986, and as far as ELT is concerned, this seems to me an exaggerated but still tenable position. And it is one that is unlikely to alter in the near future. It is possible that some schools may be able to affort to buy machines powerful enough to run educationally sound software. What seems much less probable is that anyone will find the ELT market sufficiently large to pour in the enormous funds required to write such software in the first place. Until such a time comes, it seems to me that we should be turning our attention to using the micro not as a substitute teacher, nor yet as a partner, but rather as a powerful and flexible tool.

David Eastment
June 1986

David Eastment co-ordinates Information Technology at Bell College Saffron Walden.

NOTES

1/ A recent software bibliography gives some 200 programs for the Acorn BBC in EFL. (British Council English Language Services Department: Calsoft Database, January 1986).

2/ And not just in the United States. Several recent publications in the U.K. cling to the "Programmed Instruction" approach.

See, for example: Kenning, M.J. and Kenning, M-M: An Introduction to Computer Assisted Language Learning. (Oxford University Press, 1983).

3/ Higgins, J and Johns, T: "Computers in Language Learning" (Collins, 1984).

4/ This point is cogently argued in Jones, C: "It's not so much the program....." in SYSTEM, vol.14, No.2.

5/ Self, J: "Microcomputers in Education" (pp 167). (Harvester, 1985).